

AMENDMENTS TO THE CLAIMS

1. (Original) A memory leak detection and reporting method comprising the steps of:
time stamping allocated ones of resources in a resource pool;
identifying calling code segments receiving said allocated resources;
detecting memory leaks by inspecting individual timestamps for said allocated resources
to determine whether said allocated resources have become overly idle; and,
for each allocated resource determined to have become overly idle, reporting an identity
of a corresponding one of said calling code segments.

2. (Original) The method of claim 1, wherein said identifying step comprises the step of
retrieving identities for individual ones of said calling code segments from an associated calling
stack when said individual ones of said calling code segments acquire one of said allocated
resources.

3. (Original) The method of claim 1, further comprising the step of performing said
detecting and reporting steps in a separate thread of execution.

4. (Original) The method of claim 2, wherein said retrieving step comprises the steps of:
for calling code segment in which a resource is allocated, inducing a placebo error
condition in close proximity to code for allocating said resource; and,
reading error data from said calling stack produced by said placebo error condition, said
error data comprising identity information for said calling code segment.

5. (Original) The method of claim 1, further comprising the step of performing said detecting and reporting steps responsive to allocating one of said resources in said resource pool.

6. (Original) The method of claim 1, further comprising the step of performing said detecting and reporting steps responsive to an elapsing interval.

7. (Currently Amended) A memory leak detection and reporting system comprising:
a resource pool comprising a plurality of allocable resources;
a pool manager programmed to manage allocation of said allocable resources to calling code segments; and,

a data store of allocated resources and corresponding identities for calling code segments receiving said allocated resources, wherein

the pool manager

detects memory leaks by inspecting individual timestamps for said allocated resources to determine whether said allocated resources have become overly idle; and,
for each allocated resource determined to have become overly idle, reports an identity of a corresponding one of said calling code segments to the data store.

8. (Original) The system of claim 7, wherein said pool manager comprises:
a communicative coupling to a call stack; and,
correlation logic for correlating a calling code segment reference disposed in said call stack to a concurrently allocated one of said allocable resources.

9. (Currently Amended) The system of claim 7, further comprising a garbage collector coupled to said resource pool 440.

10. (Original) A memory leak detection and reporting method comprising the steps of: allocating a resource from a resource pool, time stamping said allocated resource and recording an identity for a calling code segment acquiring said allocated resource; updating said time stamp when said allocated resource is accessed; inspecting said time stamp to determine if said allocated resource has become overly idle; and,

if it is determined that said allocated resource has become overly idle, reporting a suspected memory leak in association with said allocated resource and further reporting said recorded identity for said calling code segment which had acquired said allocated resource.

11. (Original) The method of claim 10, further comprising the step of performing said inspecting and reporting steps in a separate thread of execution.

12. (Original) The method of claim 10, further comprising the steps of performing said allocating, inspecting and reporting steps in a pool manager.

13. (Original) The method of claim 10, further comprising the step of performing said inspecting and reporting steps responsive to allocating another resource in said resource pool.

14. (Original) The method of claim 10, further comprising the step of performing said inspecting and reporting steps responsive to an elapsing interval.

15. (Original) A machine readable storage having stored thereon a computer program for memory leak detection and reporting, the computer program comprising a routine set of instructions which when executed by the machine cause the machine to perform the steps of:

time stamping allocated ones of resources in a resource pool;
identifying calling code segments receiving said allocated resources;
detecting memory leaks by inspecting individual timestamps for said allocated resources to determine whether said allocated resources have become overly idle; and,
for each allocated resource determined to have become overly idle, reporting an identity of a corresponding one of said calling code segments.

16. (Original) The machine readable storage of claim 15, wherein said identifying step comprises the step of retrieving identities for individual ones of said calling code segments from an associated calling stack when said individual ones of said calling code segments acquire one of said allocated resources.

17. (Original) The machine readable storage of claim 15, further comprising the step of performing said detecting and reporting steps in a separate thread of execution.

18. (Original) The machine readable storage of claim 16, wherein said retrieving step comprises the steps of:

for calling code segment in which a resource is allocated, inducing a placebo error condition in close proximity to code for allocating said resource; and,
reading error data from said calling stack produced by said placebo error condition, said error data comprising identity information for said calling code segment.

19. (Original) The machine readable storage of claim 15, further comprising the step of performing said detecting and reporting steps responsive to an elapsing interval.

20. (Original) A machine readable storage having stored thereon a computer program for memory leak detection and reporting, the computer program comprising a routine set of instructions which when executed by the machine cause the machine to perform the steps of:

allocating a resource from a resource pool, time stamping said allocated resource and recording an identity for a calling code segment acquiring said allocated resource;

updating said time stamp when said allocated resource is accessed;

inspecting said time stamp to determine if said allocated resource has become overly idle; and,

if it is determined that said allocated resource has become overly idle, reporting a suspected memory leak in association with said allocated resource and further reporting said recorded identity for said calling code segment which had acquired said allocated resource.